



Intel[®] Pentium[®] M Processor

Specification Update

March 2003

Notice: The Intel[®] Pentium[®] M processor may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are documented in this Specification Update.

Document Number: 252665-001



INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL® PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. Intel products are not intended for use in medical, life saving, or life sustaining applications.

Intel may make changes to specifications and product descriptions at any time, without notice.

Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them.

The Intel® Pentium® M processor may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Intel, Pentium, Celeron, Xeon, Intel SpeedStep, Intel Centrino, and the Intel logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

*Other names and brands may be claimed as the property of others.

Copyright © 2003, Intel Corporation



Contents

Revision History4

Preface5

Errata11

Specification Changes.....16

Specification Clarifications17

Documentation Changes18

Revision History

Revision Number	Description	Date
001	Initial Release	March 2003



Preface

This document is an update to the specifications contained in the following documents:

- *Intel® Pentium® M Processor Datasheet* (Order Number 252612)
- *Intel® Architecture Software Developer's Manual, Volumes 1, 2, and 3* (Order Numbers 243190, 243191, and 243192, respectively)

It is intended for hardware system manufacturers and software developers of applications, operating systems, or tools. It contains S-Specs, Errata, Documentation Changes, Specification Clarifications and Specification Changes.

Nomenclature

S-Spec Number is a five-digit code used to identify products. Products are differentiated by their unique characteristics, e.g., core speed, L2 cache size, package type, etc. as described in the processor identification information table. Care should be taken to read all notes associated with each S-Spec number.

Errata are design defects or errors. Errata may cause the Intel® Pentium® M processor's behavior to deviate from published specifications. Hardware and software, designed to be used with any given processor, must assume that all errata documented for that processor are present on all devices unless otherwise noted.

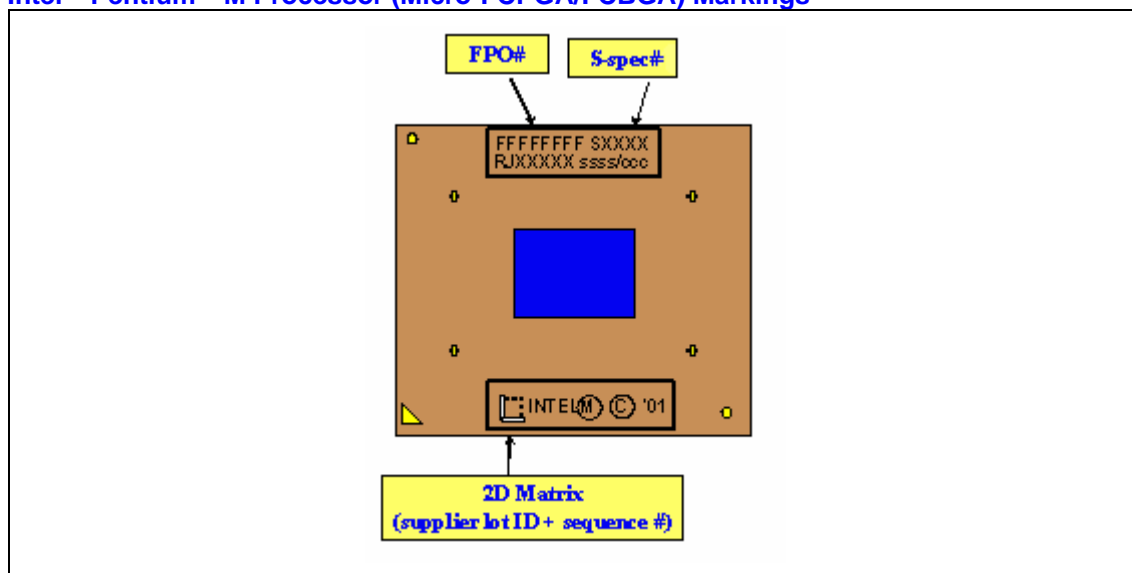
Documentation Changes include typos, errors, or omissions from the current published specifications. These changes will be incorporated in the next release of the specifications.

Specification Clarifications describe a specification in greater detail or further highlight a specification's impact to a complex design situation. These clarifications will be incorporated in the next release of the specifications.

Specification Changes are modifications to the current published specifications for the Intel® Pentium® M processor. These changes will be incorporated in the next release of the specifications.

General Information

Figure 1. Intel® Pentium® M Processor (Micro-FCPGA/FCBGA) Markings





Identification Information

The Intel® Pentium® M processor can be identified by the following values:

Family ¹	Model ²	Brand ID ³
0110	1001	00010110

NOTES:

1. The Family corresponds to bits [11:8] of the EDX register after Reset, bits [11:8] of the EAX register after the CPUID instruction is executed with a 1 in the EAX register.
2. The Model corresponds to bits [7:4] of the EDX register after Reset, bits [7:4] of the EAX register after the CPUID instruction is executed with a 1 in the EAX register.
3. The Brand ID corresponds to bits [7:0] of the EBX register after the CPUID instruction is executed with a1 in the EAX register.

Table 1, Intel® Pentium® M Processor Identification

S-Spec/ QDF	Product Stepping	CPUID	Core Speed		Bus Frequency	Package	Notes
			Highest Frequency Mode (HFM)	Lowest Frequency Mode (LFM)			
SL6N4	B-1	0695h	1.3 GHz	600 MHz	400 MHz	Micro-FCPGA	6,2
SL6F8	B-1	0695h	1.4 GHz	600 MHz	400 MHz	Micro-FCPGA	5,2
SL6F9	B-1	0695h	1.5 GHz	600 MHz	400 MHz	Micro-FCPGA	5,2
SL6FA	B-1	0695h	1.6 GHz	600 MHz	400 MHz	Micro-FCPGA	5,2
SL6N8	B-1	0695h	1.3 GHz	600 MHz	400 MHz	Micro-FCBGA	6,2
SL6F5	B-1	0695h	1.4 GHz	600 MHz	400 MHz	Micro-FCBGA	5,2
SL6F6	B-1	0695h	1.5 GHz	600 MHz	400 MHz	Micro-FCBGA	5,2
SL6F7	B-1	0695h	1.6 GHz	600 MHz	400 MHz	Micro-FCBGA	5,2
SL6NC	B-1	0695h	1.1 GHz	600 MHz	400 MHz	Micro-FCBGA	1,2
SL6NJ	B-1	0695h	900 MHz	600 MHz	400 MHz	Micro-FCBGA	3,4

NOTES:

1. VID[5:0] = 100001; VCC_CORE = 1.180 V for Highest Frequency Mode (HFM).
2. VID[5:0] = 101111; VCC_CORE = 0.956 V for Lowest Frequency Mode (LFM).
3. VID[5:0] = 101100; VCC_CORE = 1.004 V for Highest Frequency Mode (HFM).
4. VID[5:0] = 110110; VCC_CORE = 0.844 V for Lowest Frequency Mode (LFM).
5. VID[5:0] = 001110; VCC_CORE = 1.484 V for Highest Frequency Mode (HFM).
6. VID[5:0] = 010100; VCC_CORE = 1.388 V for Highest Frequency Mode (HFM).



Summary Tables of Changes

The following table indicates the Errata, Documentation Changes, Specification Clarifications, or Specification Changes that apply to Intel® Pentium® M processors. Intel intends to fix some of the errata in a future stepping of the component and to account for the other outstanding issues through documentation or specification changes as noted. This table uses the following notations:

Codes Used in Summary Table

Stepping

X: Erratum, Specification Change or Clarification that applies to this stepping.

(No mark) or (Blank Box): This erratum is fixed in listed stepping or specification change does not apply to listed stepping.

Status

Doc: Document change or update that will be implemented.

PlanFix: This erratum may be fixed in a future of the product.

Fixed: This erratum has been previously fixed.

NoFix: There are no plans to fix this erratum.

Shaded: This item is either new or modified from the previous version of the document.

Each Specification Update item is prefixed with a capital letter to distinguish the product. The key below details the letters that are used in Intel's microprocessor Specification Updates:

A = Intel® Pentium® II processor

B = Mobile Intel® Pentium® II processor

C = Intel® Celeron® processor

D = Intel® Pentium® II Xeon™ processor

E = Intel® Pentium® III processor

G = Intel® Pentium® III Xeon™ processor

H = Mobile Intel® Celeron® processor at 466/433/400/366/333/300 and 266 MHz

K = Mobile Intel® Pentium® III processor

M = Mobile Intel® Celeron® processor

N = Intel® Pentium® 4 processor

O = Intel® Xeon™ processor MP

P = Intel® Xeon™ processor

T = Mobile Intel® Pentium® 4 Processor-M

V = Mobile Intel® Celeron® processor on .13 Micron Process in Micro-FCPGA Package

W = Low Voltage Intel® Xeon™ processor

Y = Intel® Pentium® M processor

The Specification Updates for the Pentium® processor, Pentium® Pro processor, and other Intel products do not use this convention.

NO.	Stepping	Plans	ERRATA
	B1		
Y1	X	NoFix	Performance Monitoring Event that Counts Intel® Thermal Monitor 2 Transitions (59h) is not Accurate
Y2	X	NoFix	Performance Monitoring Event that counts the number of instructions decoded (D0h) is not accurate
Y3	X	NoFix	RDTSC Instruction May Report the Wrong Time Stamp Counter Value
Y4	X	NoFix	Code Segment limit violation may occur on 4 Gbyte limit check
Y5	X	NoFix	FST Instruction with Numeric and Null Segment Exceptions may cause General Protection Faults to be Missed and FP Linear Address (FLA) Mismatch
Y6	X	NoFix	Code Segment is wrong on SMM Handler when SMBASE is not aligned
Y7	X	NoFix	IFU/BSU Deadlock May Cause System Hang
Y8	X	NoFix	Processor can enter a livelock condition under certain conditions when FP exception is pending.
Y9	X	NoFix	Write Cycle of Write Combining Memory Type does not Self Snoop
Y10	X	NoFix	Performance Monitoring Event that counts Floating Point Computational Exceptions (11h) is not accurate.
Y11	X	NoFix	Inconsistent Reporting of Data Breakpoints on FP (MMX) loads
Y12	X	NoFix	Code Breakpoint may be taken after POP SS instruction if it is followed by an instruction that faults
Y13	X	NoFix	SysEnter and SysExit instructions may write incorrect Requestor Privilege Level (RPL) in the FP Code Segment selector (FCS)
Y14	X	NoFix	Memory Aliasing with Inconsistent A and D Bits may Cause Processor Deadlock

SPEC CHANGE NUMBER	STATUS	SPECIFICATION CHANGE
		There are no Specification Changes in this Specification Update revision.

NO.	DOCUMENT REVISION	STATUS	SPECIFICATION CLARIFICATIONS
			There are no Specification Clarifications in this Specification Update revision.

NO.	DOCUMENT REVISION	STATUS	DOCUMENTATION CHANGES
			There are no Documentation Changes in this Specification Update revision.

Errata

Y1. *Performance Monitoring Event that Counts Intel Thermal Monitor 2 Transitions (59h) is not Accurate*

Problem: The performance monitoring event that counts Intel Thermal Monitor 2 (Enhanced Intel SpeedStep® based) transitions may have inaccurate results.

Implication: There is no functional impact of this erratum. However this Performance Monitoring Event should not be used when accurate performance monitoring is required.

Workaround: None.

Status: For the steppings affected see the Summary of Errata at the beginning of this section.

Y2. *Performance Monitoring Event that counts the number of instructions decoded (D0h) is not accurate*

Problem: The performance-monitoring event that counts the number of instructions decoded may have inaccurate results.

Implication: There is no functional impact of this erratum. However the results/counts from this Performance Monitoring Event should not be considered as being accurate

Workaround: None.

Status: For the steppings affected see the Summary of Errata at the beginning of this section.

Y3. *RDTSC Instruction May Report the Wrong Time Stamp Counter Value*

Problem: The Time Stamp Counter is a 64-bit counter that is read in two 32-bit chunks. The counter incorrectly advances and therefore the two chunks may go out of synchronization causing the Read Time Stamp Counter (RDTSC) instruction to report the wrong time stamp counter value

Implication: This erratum may cause software to see the wrong representation of processor time and may result in unpredictable software operation.

Workaround: It is possible for BIOS to contain a workaround for this erratum.

Status: For the steppings affected see the Summary of Errata at the beginning of this section.

Y4. Code Segment limit violation may occur on 4 Gbyte limit check

Problem: Code Segment limit violation may occur on 4 Gbyte limit check when the code stream wraps around in a way that one instruction ends at the last byte of the segment and the next instruction begins at 0x0.

Implication: This is a rare condition that may result in a system hang. Intel has not observed this erratum with any commercially available software, or system.

Workaround: Avoid code that wraps around segment limit.

Status: For the steppings affected see the Summary of Changes prior to this section.

Y5. *FST Instruction with Numeric and Null Segment Exceptions may cause General Protection Faults to be Missed and FP Linear Address (FLA) Mismatch*

Problem: FST instruction combined with numeric and null segment exceptions may cause General Protection Faults to be missed and FP Linear Address (FLA) mismatch.

Implication: This is a rare condition that may result in a system hang. Intel has not observed this erratum with any commercially available software, or system.

Workaround: None.

Status: For the steppings affected see the Summary of Errata at the beginning of this section.

Y6. *Code Segment is wrong on SMM Handler when SMBASE is not aligned*

Problem: With SMBASE being relocated to a non-aligned address, during SMM entry the CS can be improperly updated which can lead to an incorrect SMM handler.

Implication: This is a rare condition that may result in a system hang. Intel has not observed this erratum with any commercially available software, or system.

Workaround: Align SMBASE to 32K byte.

Status: For the steppings affected see the Summary of Errata at the beginning of this section.

Y7. IFU/BSU Deadlock May Cause System Hang

Problem: A lockable instruction with memory operand that spans across two pages may, given some rare internal conditions, hang the system.

Implication: When this erratum occurs, the system may hang. Intel has not observed this erratum with any commercially available software or system.

Workaround: Lockable data should always be contained in a single page.

Status: For the steppings affected see the Summary of Errata at the beginning of this section.



Y8. *Processor can enter a livelock condition under certain conditions when FP exception is pending*

Problem: Processor clock modulation may be controlled via a processor register (IA32_THERM_CONTROL) or via the STPCLK# signal. While the Processor clock is constantly being actively modulated at 12.5% and 25% duty cycles and there is a pending unmasked FP exception (ES pending), if you attempt a FP load (or MMX Mov instruction) and the load has an longer than typical latency the processor can enter a livelock.

Implication: When this erratum occurs, the processor will enter a livelock condition. Intel has not observed this erratum with any commercially available software or system.

Workaround: None.

Status: For the steppings affected see the Summary of Errata at the beginning of this section.

Y9. *Write Cycle of Write Combining Memory Type does not Self Snoop*

Problem: Write cycles of WC memory type does not self-snoop. This may result in data inconsistency- if the addresses of the WC data are aliased to WB memory type memory, which has been cached. In such a case the internal caches will not be updated with the WC data sent on the system bus.

Implication: This condition may result in a data inconsistency. Intel has not observed this erratum with any commercially available software, system, nor components.

Workaround: Software should detect via the self-snoop bit in the CPUID features flags if the processor supports a self-snooping capability. Software should perform explicit memory management/flushing for aliased memory ranges on processors that do not self-snoop.

Status: For the steppings affected see the Summary of Errata at the beginning of this section.

Y10. *Performance Monitoring Event that counts Floating Point Computational Exceptions (11h) is not accurate*

Problem: Performance monitoring event that counts Floating Point Compare exceptions may have inaccurate results.

Implication: There is no functional impact of this erratum. However this Performance Monitoring Event should not be used when accurate performance monitoring is required.

Workaround: None.

Status: For the steppings affected see the Summary of Errata at the beginning of this section.

Y11. *Inconsistent Reporting of Data Breakpoints on FP (MMX) loads*

Problem: The reporting of data breakpoints on either FP or MMX loads is dependent upon the code faulting behavior prior to the execution of the load. If there is a fault pending prior to the execution of the load and FP exceptions are enabled there is a chance that data breakpoint on successive FP/MMX Loads may be reported twice.

Implication: Software debuggers should be aware of this possibility. There should be no implications to software operated outside of a debug environment.

Workaround: None.

Status: For the steppings affected see the Summary of Errata at the beginning of this section.

Y12. *Code Breakpoint may be taken after POP SS instruction if it is followed by an instruction that faults*

Problem: A POP SS instruction should inhibit all interrupts including Code Breakpoints until after execution of the following instruction. This allows sequential execution of POP SS and MOV ESP, EBP instructions without having an invalid stack during interrupt handling. However, a Code breakpoint may be taken after POP SS if it is followed by an instruction that faults, this results in a code breakpoint being reported on an unexpected instruction boundary since both instructions should be atomic.

Implication: This can result in a mismatched Stack Segment and SP. Intel has not observed this erratum with any commercially available software, or system.

Workaround: As recommended in the IA32 Intel® Architecture Software Developer's Manual the use "POP SS" in conjunction with "MOV ESP, EBP" will avoid the failure since the "Mov" will not fault.

Status: For the steppings affected see the Summary of Errata at the beginning of this section.

Y13. *SysEnter and SysExit instructions may write incorrect Requestor Privilege Level (RPL) in the FP Code Segment selector (FCS)*

Problem: SysEnter and SysExit instructions may write incorrect RPL in the FP Code Segment selector (FCS). As a result of this, the RPL field in FCS may be corrupted.

Implication: This is a rare condition that may result in a system hang. Intel has not observed this erratum with any commercially available software, or system.

Workaround: It is possible for BIOS to contain a workaround for this erratum.

Status: For the steppings affected see the Summary of Errata at the beginning of this section.



Y14. *Memory Aliasing with Inconsistent A and D Bits may Cause Processor Deadlock*

Problem: In the event that software implements memory aliasing by having two Page Directory Entries (PDEs) point to a common Page Table Entry (PTE) and the Accessed and Dirty bits for the two PDEs are allowed to become inconsistent the processor may become deadlocked.

Implication: This erratum has not been observed with commercially available software.

Workaround: Software that needs to implement memory aliasing in this way should manage the consistency of the Accessed and Dirty bits.

Status: For the steppings affected see the Summary of Errata at the beginning of this section.

Specification Changes

There are no Specification Changes in this Specification Update revision.



Specification Clarifications

There are no specification clarifications in this Specification Update revision.

Documentation Changes

There are no documentation changes in this Specification Update revision.